ARK + CheXpert

Nitin Reddy Yarava
Arizona State University
nyarava@asu.edu

Abstract

This report presents experiments conducted using the ARK deep learning model on the CheXpert dataset. The main objective was to evaluate the effectiveness of different approaches, including training from scratch and fine-tuning with pre-trained weights. The results are analyzed to understand the strengths and weaknesses of the methodologies, particularly focusing on hyperparameter selection and its impact on model performance.

1. Introduction

ARK (Accrues and Reuses Knowledge) is a deep learning model designed to learn from heterogeneous annotations across datasets. The ARK6 version is pre-trained on over 700,000 medical images, demonstrating the ability to outperform state-of-the-art (SOTA) models trained on over 800,000 proprietary images. The primary advantage of ARK lies in its capacity to effectively utilize expert-annotated heterogeneous data, making it robust across different datasets. [2]

The CheXpert dataset is a large-scale public dataset containing over 224,000 chest X-ray images spanning 14 pathologies. One of its notable challenges is the presence of uncertain labels marked as '-1'. These uncertain labels require careful handling to avoid biasing the model. In this project, the uncertain labels were treated as "LSR-Ones," meaning they were interpreted as positive (1) with label smoothing regularization (LSR). LSR prevents the model from becoming overconfident in predictions by adjusting labels to slightly less extreme values.[1]

2. Methodology

Three experimental runs were conducted with varying configurations to determine the optimal hyperparameters for training ARK on the CheXpert dataset. Each approach is outlined below, with a detailed breakdown of hyperparameters and results.

2.1. Approach 1: Training from Scratch

In the first experiment, ARK was trained from scratch on the CheXpert dataset without using any pre-trained weights.

Table 1. Hyperparameters for Approach 1: Training from Scratch

Hyperparameter	**Value**
Learning Rate	0.3
Batch Size	200
Optimizer	SGD
Epochs	10
Backbone	Swin (Base)

The rest of the model's hyperparameters were unchanged from their default values.

Results: The model achieved a test mAUC of **0.8774**, which serves as the baseline for comparison.

2.2. Approach 2: Fine-Tuning with Pre-trained Weights (ARK6)

In the second experiment, ARK was fine-tuned using pre-trained weights from ARK6. Early stopping was implemented after 10 epochs as the validation loss ceased to improve.

Table 2. Hyperparameters for Approach 2: Fine-Tuning with ARK6 Pre-trained Weights

Hyperparameter	**Value**
Learning Rate	0.01
Batch Size	64
Optimizer	SGD
Epochs	40 (Early stopping at 10)
Backbone	Swin (Base)

Results: The model achieved a test mAUC of **0.8834**, showing a slight improvement over the scratch training.

2.3. Approach 3: Fine-Tuning with ARK6 Pre-trained Weights (Modified Hyperparameters)

In the final experiment, ARK6 pre-trained weights were used again, but the optimizer was switched to AdamW, and the learning rate was reduced to 0.001. Despite these changes, the performance slightly dropped.

Table 3. Hyperparameters for Approach 3: Fine-Tuning with Modified Hyperparameters

Hyperparameter	**Value**
Learning Rate	0.001
Batch Size	64
Optimizer	AdamW
Epochs	10
Backbone	Swin (Base)

Results: The model achieved a test mAUC of **0.8555**, indicating that the modified hyperparameters were less effective.

3. Results Analysis

1. **Approach 1** (Training from scratch) provided a decent baseline, but lacked the advantages of pre-trained weights.
2. **Approach 2** demonstrated the benefit of ARK6 pre-trained weights, achieving the highest mAUC (0.8834).
3. **Approach 3** showed that changes in hyperparameters (e.g., AdamW and lower learning rate) negatively impacted performance, suggesting that fine-tuning with original defaults works better for ARK.

4. Strengths and Weaknesses

4.1. Strengths

1. **Systematic Evaluation:** The experiment tested three distinct approaches, providing a comprehensive comparison. 2. **Effective Use of Pre-training:** Fine-tuning with pre-trained weights significantly improved performance over scratch training. 3. **Uncertainty Handling:** The use of LSR-Ones for uncertain labels in CheXpert was a thoughtful and effective choice.

4.2. Weaknesses

1. **Hyperparameter Selection:** The choice of hyperparameters for Approach 3 could have been better, as the performance degraded. 2. **Limited Epochs:** While early stopping is effective, further experimentation with increased epochs and regularization might yield better results.

5. Conclusion

This report highlights the benefits of leveraging pre-trained weights for medical imaging tasks. While ARK demonstrated strong results on CheXpert, there is room for improvement in hyperparameter tuning. Future work could involve exploring advanced optimization techniques and additional regularization methods.

References

[1] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019. 1

[2] DongAo Ma, Jiaxuan Pang, Michael B Gotway, and Jianming Liang. Foundation ark: Accruing and reusing knowledge for superior and robust performance. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 651–662. Springer, 2023. 1